A Domain Specific Property Language For Fraud Detection To Support Agile Specification Development

Aaron Calafato, Christian Colombo, and Gordon J. Pace

Dept. of Computer Science University of Malta

Fraud detection is vital in any financial transaction system, including the collection of tax. The identification of fraud cases was traditionally carried out manually, having fraud experts going through their records and intuitively selecting the ones to be audited — a lengthy and unstructured process. Although work has been done with regards to the use of artificial technology for fraud pattern discovery, the results are not encouraging without major intervention by fraud experts [4].

Nowadays, in practice, patterns identified by fraud experts are coded by the software developers who select fraud cases from a database. The resulting application is verified by the fraud expert, who may feel the need to refine the rules in multiple iterations. However, this process is prone to human-induced bugs due to the continuous manual work. A better approach would include the description of rules through the use of a structured grammar, understandable by a computer system. With a compilable set of descriptions, the rules may be automatically processed against historical data — limiting the dependency on a software developer solely to the process of setting up the system.

1 Proposed Solution

In order to enable an automated system, the fraud patterns need to be translatable to a more computerised language. Fully natural languages tend to lead towards ambiguous descriptions [1]. Domain Specific Languages (DSL) [2] are languages whose expressivity is focused on the domain in question. The control of expressiveness is derived from the implementation of core concepts with further functionalities built on top of these [3]. For instance, in a tax-related fraud detection system, a rule: “Any company declaring less than 2000 Euro in profits is fraudulent", may need to be described. With a DSL, the core definitions (i) company, (ii) year, and (iii) profits would be defined first. These would subsequently be used to build rules, similar to Figure 1c. Being built on a grammar (with defined functions), a well-designed DSL reduces the ambiguity of the rules defined. For instance, the natural example does not specify the year in question, whilst in a DSL, the year may be compulsory for the definition.

DSLs like [3] however expose a syntax that is unnatural for fraud experts, since it is built on top of a programming language. Controlled Natural Languages (CNL)[5] are a type of DSL legible to human beings whilst still being easily processed by a computerised system. One of the major tools in this area is Grammatical Framework (GF) [1], which enables the definition of multiple grammars, linked through a common structure. Defining the grammars from scratch allows us to control the expressivity in the fraud detection language. We propose to translate an English-like sentence (Figure

1a) to a programming-like representation (Figure 1c), the former targeting the fraud detection patterns, whilst the latter would be used for the detection of cases. Figure 1b represents the intermediate representation between the translation. An English-like language is more natural for a fraud expert, due to the morphological inflictions done with GF. For instance, “Any company" in Figure 1a would imply a singular noun, which is subsequently reflected in the “is fraudulent" part. Once the programming code is created, an automated system can select the matching cases from a historical data and return them to the fraud expert, as shown in Figure 1d. This feedback mechanism allows for rules to be defined in an iterative style, such as refining the “2000 Euro" to another value to view the difference in the selected cases.

Fig. 1. Automated process: (a) definitions with a CNL , (b) interpreted to an abstract language, (c) translated to Java-like syntax, and (d) feedback returned to the fraud expert.

2 Conclusions

In this work we have proposed a way to allow fraud experts to define fraud patterns in a natural way while reducing possible ambiguities. This work will involve the creation of grammars, and the resulting language will be evaluated by an auditor from the Inland Revenue Department. The translated rules will then be processed, and with certain pre-processing, matched cases will be returned to the expert in a timely manner.

References

1. K. Angelov and A. Ranta. Implementing controlled languages in gf. In *Proceedings of the 2009 Conference on Controlled Natural Language*, CNL'09, pages 82–101, Berlin, Heidelberg, 2010. Springer-Verlag.
2. M. Fowler. *Domain Specific Languages*. Addison-Wesley Professional, 1st edition, 2010.
3. S. P. Jones, J. M. Eber, and J. Seward. Composing contracts: an adventure in financial engineering (functional pearl). In *ICFP '00: Proceedings of the fifth ACM SIGPLAN international conference on Functional programming*, pages 280–292, New York, NY, USA, 2000. ACM.
4. S. Maes, K. Tuyls, B. Vanschoenwinkel, and B. Manderick. Credit card fraud detection using bayesian and neural networks. In *In: Maciunas RJ, editor. Interactive image-guided neurosurgery. American Association Neurological Surgeons*, pages 261–270, 1993.
5. R. Schwitter. Controlled natural languages for knowledge representation. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1113–1121, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.